PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 5 :
H04L 9/26

A1

(11) International Publication Number: WO 94/21066

(43) International Publication Date: 15 September 1994 (15.09.94)

(21) International Application Number: PCT/AU94/00101

(22) International Filing Date: 4 March 1994 (04.03.94)

(30) Priority Data:
PL 7714 5 March 1993 (05.03.93) AU

(71) Applicant (*for all designated States except US*): TELSTRA CORPORATION LIMITED [AU/AU]; 242 Exhibition Street, Melbourne, VIC 3000 (AU).

(72) Inventor; and
(75) Inventor/Applicant (*for US only*): TAYLOR, Richard [GB/AU]; 29 Sherbrooke Lodge Road, Sherbrooke, VIC 3789 (AU).

(74) Agents: LESLIE, Keith et al.; Davies Collison Cave, 1 Little Collins Street, Melbourne, VIC 3000 (AU).

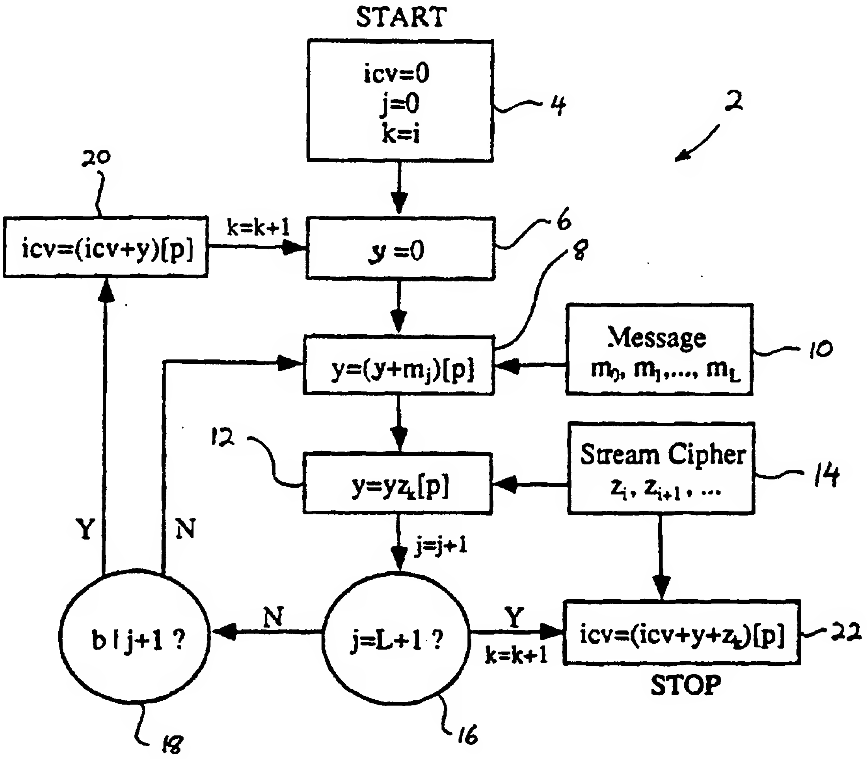
(81) Designated States: AT, AU, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, ES, FI, GB, GE, HU, JP, KG, KP, KR, KZ, LK, LU, LV, MD, MG, MN, MW, NL, NO, NZ, PL, PT, RO, RU, SD, SE, SI, SK, TJ, UA, US, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: A METHOD AND APPARATUS FOR GENERATING A DIGITAL MESSAGE AUTHENTICATION CODE

(57) Abstract

A method and apparatus for generating a message authentication code (mac) or integrity check value (icv) for a digital message to be transmitted by way of a telecommunications medium. Modular arithmetic to a prime modulus is utilised to combine message data and pseudo-random cipher data so as to produce a mac or icv which has a cryptographic strength comparable to that of the source of cipher data. The method for generating the mac can be performed iteratively, this being suitable for use with stream cipher encryption methods.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	GB	United Kingdom	MR	Mauritania
AU	Australia	GE	Georgia	MW	Malawi
BB	Barbados	GN	Guinea	NE	Niger
BE	Belgium	GR	Greece	NL	Netherlands
BF	Burkina Faso	HU	Hungary	NO	Norway
BG	Bulgaria	IE	Ireland	NZ	New Zealand
BJ	Benin	IT	Italy	PL	Poland
BR	Brazil	JP	Japan	PT	Portugal
BY	Belarus	KE	Kenya	RO	Romania
CA	Canada	KG	Kyrgystan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CG	Congo			SE	Sweden
CH	Switzerland	KR	Republic of Korea	SI	Slovenia
CI	Côte d'Ivoire	KZ	Kazakhstan	SK	Slovakia
CM	Cameroon	LI	Liechtenstein	SN	Senegal
CN	China	LK	Sri Lanka	TD	Chad
CS	Czechoslovakia	LU	Luxembourg	TG	Togo
CZ	Czech Republic	LV	Latvia	TJ	Tajikistan
DE	Germany	MC	Monaco	TT	Trinidad and Tobago
DK	Denmark	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	US	United States of America
FI	Finland	ML	Mali	UZ	Uzbekistan
FR	France	MN	Mongolia	VN	Viet Nam
GA	Gabon				

A METHOD AND APPARATUS FOR GENERATING A DIGITAL MESSAGE AUTHENTICATION CODE

This invention relates to a method and apparatus for generating a digital message authentication code.

In digital communication systems, such as broadband integrated systems digital networks (B-ISDN) it is often desirable to prevent the meaning of digital messages transmitted thereon from being intercepted and/or interfered with by an unauthorised person. For this reason, digital messages are often encrypted or enciphered such that a person intercepting the transmitted message is unable to ascertain its meaning. Therefore, at the sending site on the network a plain text message is, under control of an enciphering key, transformed into cipher text which is preferably unintelligible to anyone not having the secret deciphering key. At the legitimate receiving site on the network, the cipher text is, under control of the secret deciphering key, retransformed into the original plain text message. Cryptographic systems which operate in this way are commonly classified into block ciphers and stream ciphers.

Stream ciphers act by dividing the plain text into characters, each of which is enciphered utilising a time varying function whose time dependency is governed by the internal state of the stream cipher. The time varying function is produced by a cipher stream generator, which generates a digital cipher stream in accordance with key data which is kept secret. The cipher stream generator is constructed such that the cipher stream produced is a pseudo random bit stream which is cyclic, but has a period which is much greater than the length of key data provided. In a stream cipher, after each character is enciphered, the device changes state according to a rule, such that two occurrences of the same character in the plain text message will usually not result in the same cipher text character.

The security or strength of a stream cipher depends on the "randomness" of the generated cipher stream. Assuming an interceptor has knowledge of the plain text message, full access to the running cipher stream may also be deduced. For the system to be secure, the cipher stream must be unpredictable: regardless of the number of cipher stream digits observed, the subsequent cipher stream digits must not be more easily predictable than by just randomly guessing them. An enciphering system such as this ensures that an unauthorised person is unable to determine the meaning of an intercepted message, but does not address the issue of interference with the message despite its meaning being unknown. For example, a portion of a transmitted message may be intercepted altered or replaced with another message portion even if the interceptor is unable to ascertain the deciphered meaning of the original, altered or replaced message portion.

The immunity of a system to unauthorised and undetected alteration of a transmitted message is referred to as the integrity of the system. Integrity is a factor which is not often considered in relation to stream ciphers. A message authentication code (mac), or integrity check value (icv) determined from the content of the plain text message, may be transmitted with the cipher text to enable the receiver to determine whether the received deciphered plain text corresponds with the plain text originally transmitted, i.e. whether the cipher text has been altered during transmission. The message deciphering and authentication process involves the receiver having access to a cipher stream corresponding to the cipher stream with which the message was enciphered. Then, upon receiving the message the receiver can decipher it and generate a mac from the deciphered plain text message. A comparison of the received mac and the mac generated by the receiver can then be used as an indication of whether the transmitted mac or message has been altered in transit, since the mac generated by the receiver should be the same as the mac generated at the transmitter. However, in certain cryptographic systems it may be possible for a cryptanalyst to alter both the cipher text message and the enciphered mac in such a way that the change is not apparent to the receiver, even though the cryptanalyst is unable to determine the meaning of the cipher text which has been altered. Therefore, it is also advantageous for cryptographic systems to provide an integrity checking or authentication process which prevents such alterations during transmission from taking place without detection.

A paper entitled "A Fast Cryptographic Checksum Algorithm Based on Stream Ciphers" (X. Lai, R. Reuppel, J. Woollven; AUSCRYPT '92 Abstracts; pp 8-7 to 8-11) describes a cryptographic checksum algorithm for producing a message authentication code with a stream cipher system. The checksum algorithm presented involves demultiplexing the message stream into two subsequences according to the binary state of the cipher stream. The two subsequences are input to respective accumulating feedback shift registers, the outputs of which serve as a pair of message authentication codes. The checksum algorithm is easily implemented, regardless of the cipher stream generator structure. However, the cryptographic checksum algorithm is flawed in so far as the alteration of a single digit in the message stream only requires the alteration of a single digit in the message authentication code to obtain the correct mac value. Consequently, certain alterations can be made to the message with a high probability of also obtaining the correct message authentication code.

A further integrity checking system is described in International Patent Application No. PCT/AU93/00687 entitled "A method and apparatus for generating a cipher stream". The algorithms described therein for generating message authentication codes, however, are dependent upon the particular structure of the cipher stream generator. It would be preferable, therefore to provide an integrity checking mechanism for a stream cipher system which is independent from the method used for generating the stream cipher itself.

In accordance with the present invention there is provided a method for generating a message authentication code for a digital message in a telecommunications or computer system comprising:

generating a sequence of pseudo random cipher strings; and

generating a message authentication code by performing modular arithmetic to a prime modulus including multiplication of the digital message by a first said cipher string and addition of a second said cipher string.

Preferably, the message comprises a sequence of message units which are multiplied by respective powers of said first cipher string in generating the message authentication code.

Preferably, the message comprises a sequence of message blocks each comprising a said sequence of message units, each sequence of message units being multiplied by respective different said first cipher strings and summed with said second cipher string to form the message authentication code.

In accordance with the present invention there is also provided a method for generating a message authentication code in a telecommunications or computer system for a digital message which comprises a sequence of message blocks each comprising a sequence of message units, including the steps of:

generating a sequence of pseudo random cipher strings;

generating a non–linear function value for each message block by summing the constituent message units multiplied by respective values derived from said cipher string sequence; and

generating the message authentication code by summing the non–linear function values with a said cipher string sequence value to a prime modulus.

Preferably the message units are multiplied by respective powers of a said cipher string sequence value.

In accordance with the present invention there is also provided a method for generating a message authentication code in a telecommunications or computer system for a digital message M which comprises a sequence of message units m_j for j=0, 1, ..., r, comprising the steps of:

generating a sequence of pseudo random cipher strings z_i;

determining a non–linear function value f according to

$$f\,(M,z_i) = \sum_{x=0}^{r} m_x z_i^{r-x}\ (\mathrm{mod}\ p);\ \mathit{and}$$

generating the message authentication code Q modulus p, where p is prime, according to

$$Q = (f(M,z_i) + z_{i+1}) \pmod{p}$$

In accordance with the present invention there is also provided a method for generating a message authentication code in a telecommunications or computer system for a digital message M which comprises a sequence of message units m_j for j=0, 1, ..., r, comprising the steps of:

generating a sequence of pseudo random cipher strings z_i;

determining a non-linear function value f according to

$$f(M,z) = \sum_{x=0}^{r} m_x z_x \pmod{p}; \textit{ and}$$

generating the message authentication code Q modulus p, where p is prime, according to

$$Q = (f(M,z) + z_{r+1}) \pmod{p}$$

In accordance with the present invention there is also provided a method for generating a message authentication code in a telecommunications or computer system for a digital message M which comprises a sequence of message blocks M_j for j=0, 1, ..., s, each message block comprising a sequence of b message units m_{jk} for k=0, 1, ..., b-1, comprising the steps of:

generating a sequence of cipher strings z_j, z_{i+1}, z_{i+2} . . . z_{i+s+1};

determining a non-linear function value f for each message block according to

$$f(M_j, z_i) = \sum_{x=0}^{b-1} m_{jx} z_j^x \pmod{p}; \textit{ and}$$

generating the message authentication code Q modulus p, where p is prime, according to

$$Q(M,z_i) = (\sum_{x=0}^{s} f(M_x,z_{i+x}) + z_{i+s+1}) \pmod{p}.$$

Preferably the message authentication code effective code size is increased by a factor of h by generating a modified message authentication code Q' according to:

$$Q' = Q(M,z_i) \mid Q(M,z_{i+s+2}) \mid Q(M,z_{i+2s+4})$$
$$\ldots\ldots \mid Q(M,z_{i+(h-1)s+2}) \pmod{p}$$

where | represents concatenation.

The present invention further provides a method for encoding a digital message comprising generating a sequence of cipher strings, generating a message authentication code according to a method described above, enciphering the message by combining at least one said cipher string therewith, the at least one cipher string being distinct from the cipher strings utilised for generating the message authentication code, and appending the message authentication code to the enciphered message.

The invention also provides apparatus for generating a message authentication code for a digital message composed of a sequence of message blocks, comprising:

a stream cipher for generating a sequence of pseudo-random cipher strings; and

computation means for generating a non-linear function value for each message block by combining each message block with at least one said cipher string by way of modular arithmetic to a prime modulus, and generating a message authentication code by summing the non-linear function values together with at least one further said cipher string.

Preferred embodiments of the invention are described in detail hereinafter, by way of example only, with reference to the accompanying drawings, wherein:

Figure 1 is a flow chart of a preferred algorithm for generating a message authentication code; and

Figure 2 is a block diagram of a system for encoding digital messages for transmission by way of a telecommunications path.

An effective cipher stream generator utilises secret key data to produce an output consisting of a pseudo random bit stream Z. The cipher stream Z is typically used to encrypt a stream of message data by logically combining the cipher stream and the message stream. Since the cipher stream is continuously changing, a particular bit sequence repeated in the message stream will be encrypted differently each time, depending on the state of the cipher stream. It is therefore advantageous to exploit the time dependence randomness of the cipher stream not only for encryption of the message, but also to ensure that the integrity of the message is not compromised. Throughout the following description the terms message authenticity and message integrity are used interchangeably to refer to the condition of a digital message reaching its destination unaltered or, if altered, the alteration being detectable at the destination. In particular, the terms message authentication code (mac) and integrity check value (icv) are used interchangeably throughout the specification to denote a numerical value generated from the numerical value of a message which may be utilised to determine whether the message itself has been altered before reaching its destination. Furthermore, in the following for integers t and u we shall write t[u] to represent the unique positive integer satisfying:

$$t[u] \equiv t(\bmod\ u)\ \mathit{and}\ 0 \leq t[u] \leq (u-1)$$

Consider a stream cipher with output Z = (z_0, z_1, z_2, ...) where each output z_i is a word of w bits (typically w = 16 or 32), such that each z_i has a value from 0 to 2^w-1. It is assumed that Z is unpredictable in the sense that from any part of the Z stream it should be difficult to predict (either exactly or with high probability) what another part of the Z stream was or will be. The output of the cipher stream may be used to provide message integrity by the construction of an integrity check value (icv) that is generated from the message and appended thereto. A non-linear combination of the message and the stream cipher output is utilised to prevent an attacker from modifying the message and determining the necessary modification to generate a valid icv. Prime power modular arithmetic is also used in generating the icv, which ensures that the values of the icv are uniformly distributed and minimal in number for a given message value. The simplest icv can thus be calculated as follows, for a single message unit m_o, such as a message word, and a prime modulus p:

$$icv = (m_0z_0 + z_1)\ [p]$$

Extending this to generate a single icv for two message units yields:

$$icv = ((m_0z_o + m_1z_1) + z_2)\ [p]$$

The preferred implementation of the integrity check value generation, however, involves generating a single icv for a message which consists of a sequence of message blocks each comprising a sequence of message integer units.

The procedure for generating the icv is as follows:

Select a message block length b and prime number p = 2^w+k (k small). Let a message string M = (m_0, m_1,...), of integers between 0 and 2^w-1 be partitioned into blocks M_0, M_1,..., M_s each containing at most b integers so that:

$M_0 = m_0, m_1,..., m_{b-1}$

$M_j = m_{bj}, m_{bj+1},..., m_{b(j+1)-1}$, j = 0, 1,..., s−1

$M_s = m_{bs}, m_{bs+1},..., m_{bs+t}$, for some t ≤ b−1

Use the stream cipher to generate s+2 outputs z_i, z_{i+1}, z_{i+2},..., z_{i+s+1}. The icv is calculated as:

$$icv(M,\ b,\ p,\ z_i,\ z_{i+1},...,\ z_{i+s+1})$$

$$=(f(M_0,z_i)+f(M_1,\ z_{i+1})+...+f(M_s,\ z_{i+s})+z_{i+s+1})[p], \qquad (1)$$

where for any message string N = (n_0, n_1,..., n_r), and integer x,

$$f(N,x) = (...((n_0x+n_1)x+n_2)x+...+n_r)\ x\ [p] \qquad (2)$$

$$= (n_0x^r + n_1x^{r-1} + n_2x^{r-2} + ... +n_rx)\ [p]$$

The following example illustrates the procedure for generation of an icv for transmission with a message, such as over a telecommunications network.

Example 1: Let w = 32, p = 2^{32}+15, d = 10, b = 20. Let the cipher be in state 56 (the last output produced being z_{55}). Let M = (m_0, m_1,..., m_{108}) be a message string of length 109 that requires integrity. M is divided into blocks M_0, M_1,..., M_4 of length 20 where M_i = (m_{20i}, m_{20i+1},..., M_{20i+19}), i = 0, 1,..., 4, and one block of 9 integers M_5 = (m_{100}, m_{101},..., m_{108}). The cipher is used to generate 7 outputs z_{56}, z_{57}, z_{58},..., z_{62}, and the integrity check value is calculated according to (1) and (2).

$$icv(M,\ 20,\ 2^{32}+15,\ z_{56},\ z_{57},\ z_{58},\ldots,\ z_{62})$$

The transmitted message is then

$$m_0,\ m_1,\ldots,\ m_{108},\ icv.$$

As mentioned above, the strength of the message integrity or authentication checking system is preferably of the same order as the strength of the accompanying encryption system. In other words, the probability that an attacker is able to alter a message undetected should be comparable to the probability of the attacker successfully deciphering the message. The following Theorem and Corollaries establish a clear link between the strength of the integrity mechanism and the strength of the stream cipher from which it is constructed.

Theorem: Let p = 2^w+k, and the function f be defined by (2). Let M and M' be any two unequal message strings of length b, and y any fixed integer. Then if x is a uniformly distributed random variable in the range 0 to 2^w-1,

$$Probability[f(M,x)-f(M',x)\ \equiv\ y(\text{mod}\ p)]\leq\frac{b}{2^w}$$

Proof: Let M = (m_0, m_1,..., m_{b-1}) and M' = (m'_0, m'_1,..., m'_{b-1}). By expanding (2)

$$f(M,x)-f(M',x)\equiv((m_0\ -\ m'_0)x^b+(m_1\ -\ m'_1)x^{b-1}+\ldots+(m_{b-1}\ -\ m'_{b-1})x)(\text{mod}\ p).$$

Thus

$$f(M,x)\ -\ f(M',x)\ \equiv\ y(\text{mod}\ p)$$

if and only if

By a standard result of elementary number theory (see, for example, p58 of Ivan Niven

$$(m_0 - m'_0)x^b+(m_1 - m'_1)x^{b-1}+\ldots+(m_{b-1} - m'_{b-1})x \equiv y(\mathrm{mod}\ p).$$

and H.S. Zuckerman, *The Theory of Numbers* (fourth edition), John Wiley and Sons, 1980) such an equivalence has at most b solutions for x, from which the result follows.

Corollary 1 is an immediate consequence of this theorem.

Corollary 1: Let M and M' be any two unequal message strings, and y any fixed integer. Let the function icv() be defined as in (1) and (2). Then if $z_i, z_{i+1}, z_{i+2},\ldots, z_{i+s}$ are independent and uniformly distributed random variables in the range 0 to 2^w-1,

$$Probability[icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1})$$

$$-icv(M',\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) \equiv y(\mathrm{mod}\ p)] \leq \frac{b}{2^w}$$

Corollary 2 indicates the strength of the integrity mechanism in terms of the likelihood of replacing, in transit, a message and the corresponding icv with a legitimate, but different, message–icv pair.

Corollary 2: Let M and M' be any two unequal message strings, and y, g any fixed integers. Let the function icv() be defined as in (1) and (2). Then if $z_i, z_{i+1}, z_{i+2},\ldots, z_{i+s}, z_{i+s+1}$ are independent and uniformly distributed random variables in the range 0 to 2^w-1,

$$Probability\ [icv(M',\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) \equiv y(\mathrm{mod}\ p)$$

$$|\ icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) \equiv g(\mathrm{mod}\ p)] \leq \frac{b}{2^w} \tag{3}$$

Proof: Expanding the left hand side of the inequality above,

$$Probability[icv(M',\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) \equiv y(\mathrm{mod}\ p)$$

$$|\ icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) \equiv g(\mathrm{mod}\ p)]$$

However

$$= \mathit{Probability}[icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) - icv(M',\ b,\ p,\ z_i,\ z_{i+1},\ldots,z_{i+s+1})$$

$$\equiv g\ -\ y(\bmod p)\ \mid\ icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1})\ \equiv g(\bmod p)].$$

$$icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) - icv(M',\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1})$$

$$\equiv (f(M_0,\ z_i)+f(M_1,\ z_{i+1})+\ldots+f(M_s,\ z_{i+s})-f(M'_0,\ z_i)+f(M'_1,\ z_{i+1})-\ldots-f(M'_s,\ z_{i+s}))(\bmod p)$$

is independent of z_{i+s+1} while

$$icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1})\ =\ g\ (\bmod p)$$

if and only if

$$z_{i+s+1}\ \equiv\ (g - f(M_0,\ z_i) - f(M_1,\ z_{i+1}) - \ldots - f(M_s,\ z_{i+s}))(\bmod p).$$

Thus the events described in the conditional probability of (3) are independent and so the left hand side of (3) is equal to

$$\mathit{Probability}[icv(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+s+1}) - icv(M',\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_i + s+1)$$

$$\equiv (g-y)(\bmod p)].$$

The result now follows by Corollary 1.

Assume that the stream cipher produces output that are independent and uniformly distributed random variables in the range 0 to 2^w-1. It follows from Corollary 2 that if any message and its integrity check value were to be altered in transit (the message being altered in at least one bit position), the new message and integrity check value would register as valid by the receiver with probability at most $b/2^w$. Thus we refer to $\log_2 (2^w/b)$ as the *effective icv size*. As an example, with a word size w of 32 bits and a block size b of 20 the resulting effective icv size is approximately 28. Note that the effective icv size indicates the strength of the integrity method used with an idealised stream cipher (with outputs that are uniformly distributed independent random variables). For this to be a meaningful indicator of integrity strength with a practical deterministic stream cipher however, clearly the stream cipher key size must be at least as large as the effective icv size. In this case if there is some way of altering or substituting message–icv pairs that goes undetected with a probability of more than $b/2^w$ then this implies some corresponding level of predicability in the stream cipher output.

Figure 1 illustrates a flow chart of the preferred method of generating an integrity check value (icv) or message authentication code (mac). The flow chart 2 begins with an initialisation step 4 in which the icv and indices j and k are initialised such that icv = 0, j = 0 and k = i, where i is the initial state of the stream cipher with output z. In order to determine the icv for a given message according to equations (1) and (2), the flow chart 2 is structured into a dual loop iterative process, wherein steps 6, 8, 12 and 18 are used for the calculation of non linear function y (equation (2)) whilst steps 16, 20 and 22 perform the steps necessary for the calculation of the icv according to equation (1). Step 6 acts as an initialisation step for the non linear function value y, such that the first iteration of steps 8 and 12 are effective to calculate the first term of equation (2) (ie: n_0x). For each iteration of steps 8 and 12 successive integer values of the message are input from the message stream 10 and values of the stream cipher are input from cipher stream 14 according to the message block counter index k. Following each iteration of step 12 the index j is incremented by one, and a test of the value of index j is applied at step 16 to determine whether or not the last unit of the message has been processed. Where there remains message units left to process, the procedure continues to step 18 where a test is applied to index j to determine whether or not the end of a message block has been reached. If j+1 is an integer divisor of block size b at step 18, this indicates that the beginning of a new message block has been reached, and the procedure continues to step 20 where the icv is incremented by the value of the non linear function y. The block counter index k is incremented following step 20, and the procedure passes to step 6 where the non linear function y is reset. Where the beginning of a message block has not been reached at step 18 the procedure returns to step 8, so as to perform a further iteration of steps 8, 12 and 16. When the end of the message has been reached, as indicated by the result at step 16, the block counter index k is again incremented and the icv is totalled together with the final stream cipher value z_k (step 22). Utilising this method allows the icv to be calculated in a serial manner, which is consistent with the continuous output form of the message stream and cipher stream.

The following describes a modification of the above described integrity system, to enable the effective icv size to be increased by any required factor h. As above, a message string M = (m_0, m_1,...) is divided into blocks M_0, M_1,..., M_s each containing at most b integers. The stream cipher is used to generate h(s+2) outputs z_i, z_{i+1}, z_{i+2}, ..., $z_{i+h(s+2)-1}$. Then the integrity check value icv_h is defined as a sequence of h integers between 0 and p calculated according to:

$$icv_h(M,\ b,\ p,\ z_i,\ z_{i+1},\ldots,\ z_{i+h(s+2)-1})$$

$$=(f(M_0,\ z_i)\ +\ f(M_1,\ z_{i+1})+\ldots+f(M_s,\ z_{i+s})+z_{i+s+1})[p],$$
$$(f(M_0,\ z_{i+s+2})\ +\ f(M_1,\ z_{i+s+3})+\ldots+f(M_s,\ z_{i+2s+2})\ +\ z_{i+2s+3})\ [p],$$

$$\bullet$$

$$\bullet$$

$$(f(M_0,\ z_{i+(h-1)(s+2)})+f(M_1,\ z_{i+(h-1)(s+2)+1})+\ldots+f(M_s,\ z_{i+h(s+2)-2})+z_{i+h(s+2)-1})[p]$$

where the function f is given by (2). This provides an icv of length hp with an effective icv size of

$$\log_2\left(\left(\frac{2^w}{b}\right)^h\right)$$

Thus, for example, if b = 20, w = 32, h = 4 then the effective icv size is

$$4\ (32\ -\ \log_2 20)\approx\ 110.7$$

To provide both integrity and encryption the cipher stream can be used to provide input to the integrity calculation as well as output to be used for message encryption. In order to prevent the integrity mechanism from being undermined by a known plain text attack it is important that cipher stream output that is used in icv calculations by the receiver never be used for message encryption by the sender. Otherwise a known plaintext attack combined with altering the synchronisation of the cipher stream may succeed in making the receiver use cipher data in icv calculations which is known to an attacker. To overcome this problem an integer d (< b) is chosen such that only those z_i for which i is a multiple of d are used in the icv calculation, the remaining z_i being used for encryption. This technique is illustrated in the example below.

Example 2: As in Example 1 let w = 32, p = 2^{32}+15, d = 10, b = 20, M = (m_0, m_1,..., m_{108}) and the cipher be in state 56. To apply integrity and confidentiality the cipher is used to generate 121 outputs z_{56}, z_{57}, z_{58},..., z_{176}, and the icv is calculated as,

$$icv(M,\ 20,\ 2^{32}\ +15,\ z_{60},\ z_{70},...,\ z_{120}),$$

where the icv function is defined by (1) and (2). The transmitted message is

$$(m_0\ +\ z_{56})[2^{32}],\ (m_1\ +\ z_{57})[2^{32}],...,\ (m_4\ +\ z_{61})[2^{32}],...,\ (m_{108}\ +\ z_{176})[2^{32}],\ icv.$$

Figure 2 illustrates a simple block diagram of an encryption and integrity system which may be utilised to implement the above described. Message data m_j is output from a message source 26, which message data is passed to both an encryption processor 30 and a message authentication code generator 32. A cipher stream generator 28 outputs stream cipher data z_i, which also passes to both the encryption processor 30 and mac generator 32. The encryption processor 30 acts to combine the message and cipher stream data so as to encrypt the message for transmission thereof. Meanwhile, the mac generator 32 produces an integrity check value, as described above, utilising the same message data but only one of out every d cipher stream outputs, the other d–1 cipher stream outputs being utilised for encryption by the encryption processor 30. The encrypted message m'_j and the icv are passed to a transmission source 34 whereat the icv is appended to the encrypted message for transmission on an output 36.

The foregoing detailed description has been put forward merely by way of explanation only, and is not intended to be limiting to the invention, which is defined in the claims appended hereto.

GLOSSARY

Z = pseudo–random cipher stream sequence of cipher strings z_j;

z_i = cipher string word of w bits (z_i is an integer in the range 1 to (2^w-1))

M = digital message

= $(m_0, m_1, m_2, ...)$ sequence of integer words in the range 0 to (2^w-1)

= $(M_0, M_1, ..., M_s)$ sequence of message blocks of length b, such that message length $L \leq (s+1)\,b$, where:

M_0 = $m_0, m_1, ..., m_{b-1}$

M_j = $m_{bj}, m_{bj+1}, ..., m_{b(j+1)-1}$

M_s = $m_{bs}, m_{bs+1}, ..., m_{bs+t}$ for $t \leq (b-1)$

icv = integrity check value

$t[u]$ = $t(\text{mod } u)$

p = $2^w + k$ (k small) such that p is prime

$f(N,x)$ = $n_0x^r + n_jx^{r-1} + n_2x^{r-2} + ... + n_rx\ [p]$

= $(...((n_0x + n_1)\,x + n_2)\,x + ... + n_r)\,x\ [p]$

for $N = n_0, n_1, n_2, ... , n_r$

and integer word x

$icv\,(M, b, p, Z)$ = $(f(M_0, Z_i) + f(M_1, z_{i+1}) + ...$

$+ f(M_s, z_{i+s}) + z_{i+s+1})\ [p]$

CLAIMS:

1. A method for generating a message authentication code for a digital message in a telecommunications or computer system comprising:

generating a sequence of pseudo random cipher strings; and

generating a message authentication code by performing modular arithmetic to a prime modulus including multiplication of the digital message by a first said cipher string and addition of a second said cipher string.

2. A method as claimed in claim 1, wherein the message comprises a sequence of message units which are multiplied by respective powers of said first cipher string in generating the message authentication code.

3. A method as claimed in claim 1, wherein the message comprises a sequence of message blocks which are each multiplied by respective different first cipher strings in generating the message authentication code.

4. A method as claimed in claim 2, wherein the message comprises a sequence of message blocks each comprising a said sequence of message units, each sequence of message units being multiplied by respective different said first cipher strings and summed with said second cipher string to form the message authentication code.

5. A method as claimed in any preceding claim wherein a plurality of message authentication codes are generated for the same message but utilising different cipher strings, and the plurality of message authentication codes combined or concatenated to form a further message authentication code.

6. A method for generating a message authentication code in a telecommunications or computer system for a digital message which comprises a sequence of message blocks each comprising a sequence of message units, including the steps of:

generating a sequence of pseudo random cipher strings;

generating a non–linear function value for each message block by summing the constituent message units multiplied by respective values derived from said cipher string sequence; and

generating the message authentication code by summing the non-linear function values with a said cipher string sequence value to a prime modulus.

7. A method as claimed in claim 6, wherein the message units are multiplied by respective powers of a said cipher string sequence value.

8. A method as claimed in claim 6, wherein in generating each non-linear function value the message units are multiplied by respective powers of a said cipher stream sequence value, a different sequence value being utilised for each non-linear function value.

9. A method as claimed in claim 8, wherein the cipher string sequence value summed with the non-linear function values to generate the message authentication code is a different sequence value from the cipher strings used to generate the non-linear function values.

10. A method as claimed in any one of claims 6 to 9, wherein the non-linear function values, f, are generated according to:

$$f(M,z_i) = \sum_{x=0}^{r} m_x z_i^{r-x} \pmod{p}$$

where M is a message block comprising r message units m_x (x=0,1,....,r), and z_i are said cipher string sequence values.

11. A method for generating a message authentication code in a telecommunications or computer system for a digital message M which comprises a sequence of message units m_j for j=0, 1, ..., r, comprising the steps of:

generating a sequence of pseudo random cipher strings z_i;

determining a non-linear function value f according to

$$f\,(M,z_i) = \sum_{x=0}^{r} m_x z_i^{r-x} \text{ (mod } p\text{); } \textit{and}$$

generating the message authentication code Q modulus p, where p is prime, according to

$$Q = (f(M,z_i) + z_{i+1}) \text{ (mod } p\text{)}$$

12. A method as claimed in claim 11, wherein the message is composed of a sequence of message blocks M_k which each comprise a said sequence of message units, and wherein the message authentication code Q, modulus p, is generated according to:

$$Q = \left(\sum_{x=0}^{k} f\,(M_x,\, z_{i+x}) + z_{i+k+1}\right) \text{ (mod } p\text{)}$$

13. A method for generating a message authentication code in a telecommunications or computer system for a digital message M which comprises a sequence of message units m_j for j=0, 1, ..., r, comprising the steps of:

generating a sequence of pseudo random cipher strings z_i;

determining a non–linear function value f according to

$$f(M,z) = \sum_{x=0}^{r} m_x z_x \text{ (mod } p\text{); } \textit{and}$$

generating the message authentication code Q modulus p, where p is prime, according to

$$Q = (f(M,z) + z_{r+1}) \text{ (mod } p\text{)}$$

14. A method for generating a message authentication code in a telecommunications or computer system for a digital message M which comprises a sequence of message blocks M_j for j=0, 1, ..., s, each message block comprising a sequence of b message units m_{jk} for k=0, 1, ..., b–1, comprising the steps of:

generating a sequence of cipher strings z_i, z_{i+1}, z_{i+2} . . . z_{i+s+1};

determining a non-linear function value f for each message block according to

$$f(M_j, z_i) = \sum_{x=0}^{b-1} m_{jx} z_j^x \text{ (mod } p\text{); } \textit{and}$$

generating the message authentication code Q modulus p, where p is prime, according to

$$Q(M,z_i) = (\sum_{x=0}^{s} f(M_x,z_{i+x}) + z_{i+s+1}) \text{ (mod } p\text{).}$$

15. A method as claimed in claim 14, including the step of increasing the effective code size of the message authentication code by a factor of h by generating a modified message authentication code Q' according to:

$$Q' = Q(M,z_i) \,|\, Q(M,z_{i+s+2}) \,|\, Q(M,z_{i+2s+4})$$

$$.... \,|\, Q(M,z_{i+(h-1)s+2}) \qquad \text{(mod } p\text{)}$$

where | *represents concatenation.*

16. A method as claimed in any one of claims 1, 6, 11, 13 or 14 wherein the sequence of cipher strings comprises a subset selection of cipher string values from a cipher stream.

17. A method as claimed in claim 16 wherein the remaining cipher string values from the cipher stream are utilised for encrypting the message and/or the message authentication code.

18. A method for encoding a digital message comprising generating a sequence of cipher strings, generating a message authentication code according to any one of claims 1, 6, 11, 13 or 14, enciphering the message by combining at least one said cipher string therewith, the at least one cipher string being distinct from the cipher strings utilised for generating the message authentication code, and appending the message authentication code to the enciphered message.

19. Apparatus for generating a message authentication code for a digital message composed of a sequence of message blocks, comprising:

a stream cipher for generating a sequence of pseudo–random cipher strings; and

computation means for generating a non–linear function value for each message block by combining each message block with at least one said cipher string by way of modular arithmetic to a prime modulus, and generating a message authentication code by summing the non–linear function values together with at least one further said cipher string.

20. Apparatus according to claim 19, including:

encryption means for encrypting the message by utilising sequence values of said pseudo–random cipher string sequence which are distinct from the sequence values used to generate the message authentication code; and

means for appending the message authentication code to the encrypted message for transmission thereof.

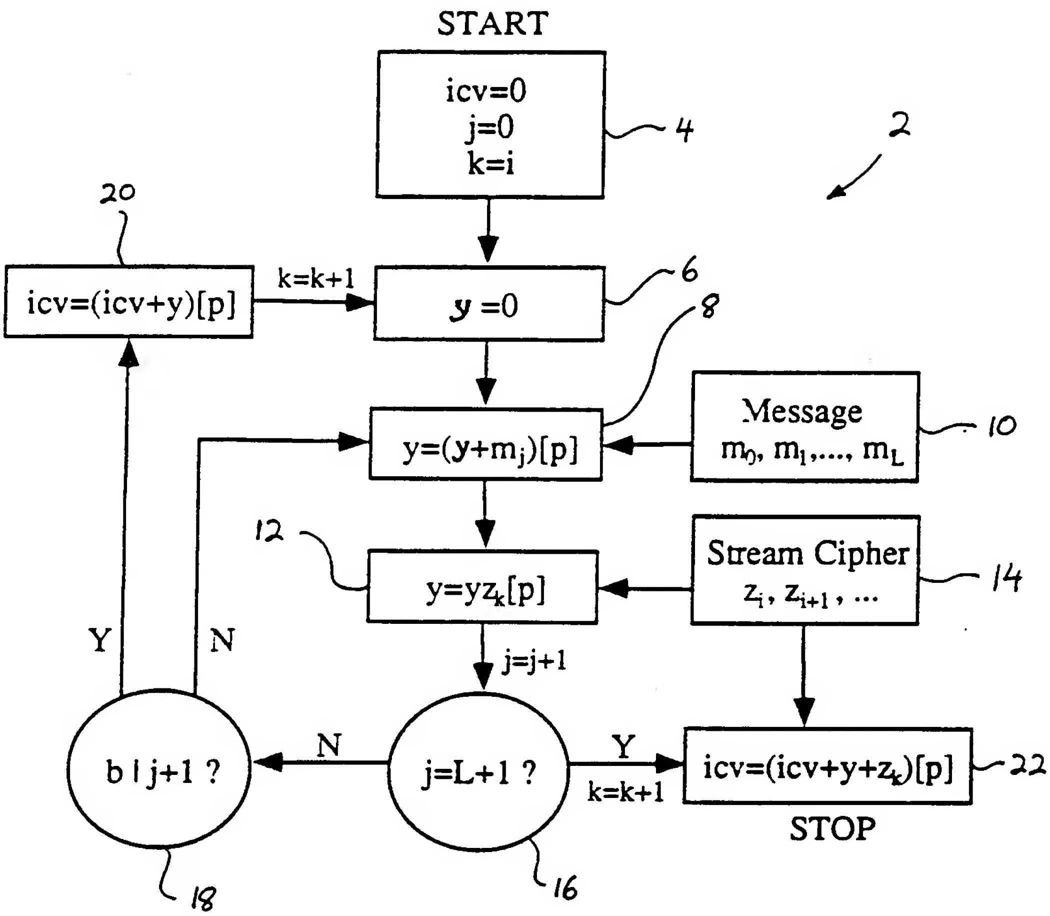


Figure 1

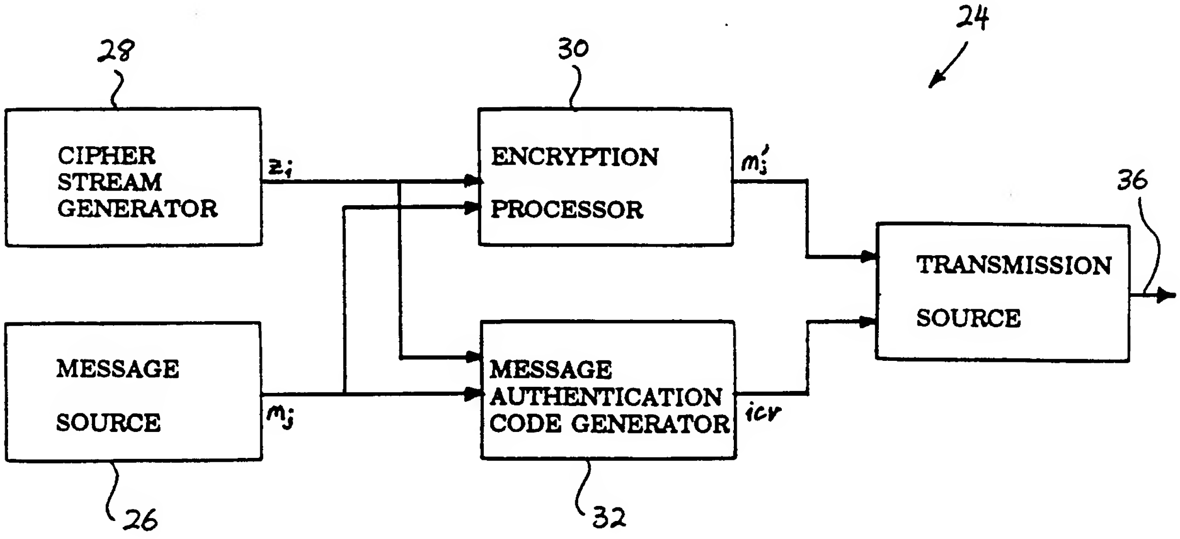


Figure 2

INTERNATIONAL SEARCH REPORT

International application No.
PCT/AU 94/00101

A. CLASSIFICATION OF SUBJECT MATTER

Int. Cl.^{5} H04L 9/26

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC H04L 9/02, 9/04. 9/26, 9/32

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
AU : IPC as above

Electronic data base consulted during the international search (name of data base, and where practicable, search terms used)
PLUS, JOPAL.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to Claim No.
A	US,A, 5146500 (MAURER) 8 September 1992 (08.09.92)	(1-20)
A	US,A, 3657476 (AIKEN) 18 April 1972 (18.04.72)	(1-20)
A	US,A, 4972474 (SABIN) 20 November 1990 (20.11.90)	(1-20)

□ Further documents are listed in the continuation of Box C.

□ See patent family annex.

* Special categories of cited documents :

"A"	document defining the general state of the art which is not considered to be of particular relevance
"E"	earlier document but published on or after the international filing date
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O"	document referring to an oral disclosure, use, exhibition or other means
"P"	document published prior to the international filing date but later than the priority date claimed
"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&"	document member of the same patent family

Date of the actual completion of the international search 18 April 1994 (18.04.94)	Date of mailing of the international search report 21 April 1994 (21.04.94)
Name and mailing address of the ISA/AU AUSTRALIAN INDUSTRIAL PROPERTY ORGANISATION PO BOX 200 WODEN ACT 2606 AUSTRALIA Facsimile No. 06 2853929	Authorized officer a. W. Duke **A.W. DUKE** Telephone No. (06) 2832174

This Annex lists the known "A" publication level patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent Document Cited in Search Report		Patent Family Member	
US	5146500	EP	503119
US	4972474		
US	3657476		
			END OF ANNEX

PUB-NO:	WO009421066A1
DOCUMENT-IDENTIFIER:	WO 9421066 A1
TITLE:	A METHOD AND APPARATUS FOR GENERATING A DIGITAL MESSAGE AUTHENTICATION CODE
PUBN-DATE:	September 15, 1994

INVENTOR-INFORMATION:

NAME	**COUNTRY**
TAYLOR, RICHARD	AU

ASSIGNEE-INFORMATION:

NAME	**COUNTRY**
TELSTRA CORP LTD	AU
TAYLOR RICHARD	AU

APPL-NO: AU09400101

APPL-DATE: March 4, 1994

PRIORITY-DATA: AUPL771493A (March 5, 1993)

INT-CL (IPC): H04L009/26

EUR-CL (EPC): H04L009/32

US-CL-CURRENT: 713/FOR.123

ABSTRACT:

A method and apparatus for generating a message authentication code (mac) or integrity check value (icv) for a digital message to be transmitted by way of a telecommunications medium. Modular arithmetic to a prime modulus is utilised to combine message data and pseudo-random cipher data so as to produce a mac or icv which has a cryptographic strength comparable to that of the source of cipher data. The method for generating the mac can be performed iteratively, this being suitable for use with stream cipher encryption methods.